2005 Asia-Pacific Conference on Communications, Perth, Western Australia, 3 - 5 October 2005.

Low Variance Loss Event Rate Estimation for Layered Multicast Protocol

Osman Ghazali* and Suhaidi Hassan†

Department of Computer Sciences, Faculty of Information Technology
Universiti Utara Malaysia, 06010 Sintok, Kedah, Malaysia
E-mail: *osman@uum.edu.my , †suhaidi@uum.edu.my

***Abstract*—In a high level of statistical multiplexing environment, the packet drop pattern of a data flow is determined by the aggregate data flow behaviour of the bottleneck link. However, an observation of loss event patterns of a layered multicast session shows oscillatory loss intervals are estimated at receivers. Under this condition, the current loss event rate estimation technique of layered multicast protocols estimates oscillatory loss event rates. Moreover, the protocols inability to synchronise packet sequence number across multicast layers exaggerates this problem. To address this problem, we propose Two-step loss interval filtering technique that discards the unrepresentative loss interval samples. The proposed loss interval filtering technique reduces the variance of estimated loss event rates and results in stable protocol behaviour.**

Index Terms - **Congestion Control, Loss Rate, Layered Multicast, Transport Protocol**

I. INTRODUCTION

Layered Multicast Protocol (LMP) is regarded as one of the solutions for data transmission of continuous multi-media applications over the best-effort Internet services. LMP allows users with different network capacities to achieve different reception rates and therefore users of different network bandwidth perceive different multimedia qualities.

TCP-equation model is the mechanism commonly used to control congestion in TCP-compatible rate-based layered multicast protocols. A typical TCP equation models the steady state of TCP throughput with the functions of packet size, loss event rate (LER), round-trip time and retransmission timeout. It has been adopted in many non-TCP protocols as it enables the protocols to control congestion and at the same time to be friendly towards TCP data flows.

LER, that is the inverse of loss interval size, is regarded as one of the most important parameters in the TCP-equation model [1]. In comparison with other parameters, it has greater influence on the accuracy and stability of TCP-compatible rate estimations. However, due to oscillatory estimated loss intervals, oscillatory TCP-compatible rates are estimated [2]. In a layered multicast session, this problem is exaggerated by the misleading loss event information, which is the result of the inability of the sender of a layered multicast session to assign the session's sequence numbers to the packets. To address this problem, we propose Two-step lost interval filtering technique that discards the too high or too low unrepresentative loss interval samples. The proposed filtering technique reduces the variance of the estimated loss intervals and the estimated LERs, and results in stable protocol behaviour.

The remainder of this paper is organised as follows. The next section gives a brief overview of the TCP-friendly equation model, Section III gives an overview and describes the problem of LER estimation, Section IV proposes Two-step loss interval filtering technique, Section V describes One-step loss interval filtering technique, Section VI describes the experiment settings, Section VII presents the results, Section VIII discusses the results, and Section IX concludes this paper.

II. TCP- FRIENDLY EQUATION MODEL

TCP is the most dominant traffic in the Internet, hence it is suggested [1] that other protocols have to be friendly towards TCP. Equation-based LMPs employ the TCP-equation model as the mechanism to control congestion, and to be friendly towards TCP data flows. A number of TCP-friendly equations that model TCP steady-state throughput have been proposed, and the most popular model is the TCP Reno equation model proposed by [3]. Using the model, equation-based LMPs can estimate a TCP-compatible rate, and adjust their sending or reception rate based on the estimated TCP-compatible rate.

III. LER ESTIMATION

LER is suggested as the better representation of general TCP behaviour [3]. It is the inverse of the size of a loss interval, and the size of a loss interval is the number of received and lost packets within a loss interval. A loss interval begins with a loss event and ends with another loss event. A loss interval may contain one or more packet loss occurrences during one round trip time. A lost packet is considered a part of an existing loss interval if it occurs within a RTT since the last loss event. Otherwise, the lost packet becomes the first packet of a new loss event.

A. The Problem of LER Estimation

In a high level of statistical multiplexing environment, the packet drop pattern of a data flow is determined by the aggregate data flow behaviour of the bottleneck link. An observation of loss event patterns of a layered multicast protocol shows oscillatory loss intervals are estimated at receivers [2]. Though average loss interval algorithm [3] is used to mitigate the effect of the oscillatory loss intervals, the small size of loss history windows limits the effectiveness of this technique. Therefore oscillatory LERs are estimated at receivers.

In a LMP session, data packets are distributed across multicast layers where each layer can be seen as a single layered multicast. Therefore, it is not possible to assign the session's sequence numbers to the packets. Consequently, the packets can only be assigned layers' sequence numbers. However, the assigned layers' sequence numbers mislead receivers regarding the packet loss events and the size of lost intervals. As a result, LERs are wrongly estimated at receivers. This problem is likely to affect the LMPs with a layering scheme of high rate multipliers more than the LMPs with a layering scheme of low rate multipliers. That is, the higher the rate multiplier, the larger the effect of misleading information from the layers' sequence number.

A number of layering schemes are used in the current LMPs. Layering schemes that imitate AIMD behaviour of TCP such as in [4] use a rate multiplier of 2, while protocols such as in [5] recommend a lower rate multiplier of 1.3. Some other protocols use dynamic layering schemes that adjust layers' size according to the present network status.

B. Average Loss Interval

An average loss interval algorithm is recommended as the best weighted average for LER estimations [3]. The algorithm uses a dynamic history window and the exponential weighted moving average. An average loss interval size is computed as the weighted average of the last *k* loss intervals as follows:

$$l_{avg}(n) = \frac{\sum_{i=0}^{k-1} w_i l_{n-1}}{\sum_{i=0}^{k-1} w_i} \qquad (1)$$

and for weights w_i:

$$w_i = \begin{cases} 1 & \text{for} \quad 1 < i < n/2, \\ 1 - \dfrac{i - n/2}{n/2+1} & \text{for} \quad n/2 < i < n. \end{cases} \qquad (2)$$

The recommended windows size is k=8, which gives weights of 1, 1, 1, 1, 0.8, 0.6, 0.4, and 0.2 for w_1 through w_8 respectively.

IV. Two-step Loss Interval Filtering Technique

The oscillatory packet drop pattern and the misleading packet sequence numbers result in fluctuations of observed loss interval sizes at layered multicast receivers. This results in some of the observed loss interval sizes are too high or low, and are not representing the actual network conditions. These unrepresentative loss intervals are temporary, that they will not form a new loss interval trend.

With the assumption that unrepresentative loss interval changes are temporary, we proposed Two-step loss interval filtering technique that identifies and discards the temporary extreme changes of observed loss interval size - only the observed loss intervals that are within the current loss interval trend are considered for inclusion in the loss windows history. The technique consists of a preliminary test and two filtering steps. The preliminary test is to examine the newly observed loss interval and assign its status, the first filtering step is to test whether the change in the observed loss interval is the formation of a new loss interval trend, and the third step is to confirm the formation of the new loss interval trend.

To implement Two-step loss interval filtering technique, two additional loss interval history windows are required on top of the recommended loss history windows - so the new size of loss history windows is k=8+2=10. The additional windows are for the placement of conditional loss interval samples. Their statuses are conditional because the acceptances in the loss history windows are subject to the confirmation of the new loss interval trend. A newly observed loss interval that enters the loss history windows will be place in the first window; this window has no weight in the average loss interval calculation as its pattern is not confirmed as yet. The second window is given weights of 1 in average loss interval calculation as it passed the first step filtering, though it still requires confirmation of the new trend formation.

A. Preliminary Test

A newly observed loss interval that is outside of the range of the current loss interval trend is considered as an indicator of the formation of a new loss interval trend.

For the preliminary test, we set the upper boundary (*UB*) of the loss interval trend using (3), while the lower boundary (*LB*) of the loss interval trend is set using (4). The rate factor (α) can be set between 0 and 1.0. The nearer α is to 1.0, the wider the range of the current trend, while the nearer α is to 0, the narrower the range of the current trend. In our experiments, we observe a right-skewed loss-event rate distribution. We also observe that the right tail of the loss event rate distribution is very long. Considering these, α is set to 0.5, but the rate factor is doubled for the *UB*.

$$UB = l_{avg} + 2\alpha . l_{avg} \qquad (3)$$

$$LB = l_{avg} - \alpha . l_{avg} \qquad (4)$$

Whereupon a new loss interval sample is observed, we examine the newly observed loss interval sample by comparing its size with the range of the current loss interval trend. If the newly observed loss interval is within the range of the current trend, it will be accepted for the inclusion in the loss interval history windows and is assigned status 1. On the other hand if it is outside of the range of the current loss interval trend, it is assigned status 2 if it is above the range of the current loss interval trend, and it is assigned status 3 if it is below the range of the current loss interval trend. A status of 2 and 3 means the acceptance for the inclusion in the loss interval history windows is subject to confirmation of the new loss interval trend. The loss interval sample with any of these statuses will be further tested when a new loss interval is observed. Thus at this stage, the loss interval sample is placed into the first window (l_1) and is given no weight in the average loss interval calculation. The preliminary test algorithm is outlined below.

Preliminary Test
IF l_{sample} >= LB AND l_{sample} <= UB
 $l_1 = l_{sample}$
 $status_1 = 1$
ELSEIF $l_{sample} > LB$
 $l_1 = l_{sample}$
 $status_1 = 2$
ELSEIF $l_{sample} < UB$
 $l_1 = l_{sample}$
 $status_1 = 3$
END

B. *Step One*

Step one process is triggered in the event of a new loss, and another loss interval sample is observed. This step will further test the LER sample from the preliminary test. This step is to check the formation of a new loss interval trend by examining whether the loss interval sample is within the l_1 trend. If the new loss interval sample is within the l_1 trend, the l_1 is accepted for moving to window 2 (l_2) of the loss interval history windows and it carries the same status number. Otherwise l_2 will be assigned the current average loss interval and assigned status number 1. The step-one test algorithm is outlined below.

Step One
IF $status_1 = 1$
 $l_2 = l_1$
 $status_2 = 1$
ELSEIF $status_1 = 2$ AND $l_{sample} > (1-\alpha).\ l_1$
 $l_2 = l_1$
 $status_2 = 2$
ELSEIF $status_1 = 3$ AND $l_{sample} < (1+\alpha).\ l_1$
 $l_2 = l_1$
 $status_2 = 3$
ELSE
 $l_2 = l_{avg}$
 $status_2 = 1$
END

C. *Step Two*

Step two is the final step and it is to further confirm the formation of the new loss interval trend. Similar to previous steps, it is triggered in the event of a new loss and a new loss interval is observed. This step will examine whether the new loss event sample is within the trend range of the l_2. If it is within the trend range of the l_2, the l_2 is accepted for move to window 3 (l_3), which means it is fully accepted for the inclusion in the loss history windows. If the new loss event sample is not within the trend range of the l_2, the l_2 will not be accepted for the inclusion in the loss history windows, and l_3 will be assigned the current average loss interval. The step-two test algorithm is outlined below.

Step Two
IF $status_2 = 1$
 $l_3 = l_2$
ELSEIF $status_2 = 2$ AND $l_{sample} > (1-\alpha).\ l_2$
 $l_3 = l_2$
ELSEIF $status_2 = 3$ AND $l_{sample} < (1+\alpha).\ l_2$
 $l_3 = l_2$
ELSE
 $l_3 = l_{avg}$
END

V. One-step Loss Interval Filtering Technique

One-step loss interval filtering technique is similar to Two-step loss interval filtering technique but performs less loss interval filtering tests. It does not perform the confirmation of the new loss interval trend formation, where it stops at step one of Two-step loss interval filtering technique. With this technique, the loss interval sample that passed the step one filtering will be accepted for the inclusion in the loss history windows. Since it performs fewer testing steps than Two-step loss interval filtering, it also requires lesser loss history windows. That is, it requires 9 loss history windows compared to 10 windows in Two-step loss interval filtering technique.

VI. Experiments

To evaluate the loss interval filtering technique, we implement the technique in a TCP-Friendly Layered Multicast Protocol (TFLMP) [6] with fixed sending and reception rate. Then we set experiments with the objectives to study the stability and precision of estimated LERs and TCP-compatible rates.

D. *Performance Metric*

To evaluate the precision of LER and TCP-compatible rate estimations, we compare the estimated LERs and the resultant TCP-compatible rates with the theoretical fair bandwidth share per-data-flow, i.e. 200 Kbps per-data-flow. Coefficient of variation (CoV) and variability measurement as in [7] are used to evaluate the stability of LER and TCP-compatible rate estimations. The calculation of CoV and variability are performed on a per-data-flow basis, the per-data-flow results are averaged for all simulations.

E. *Simulation Setting*

Three different TFLMPs are used in this study. The first is a TFLMP that employs the LER estimation technique similar in [8], which is labelled TFLMP-1. The second is the TFLMP-1 with One-step loss interval filtering technique, which is labelled TFLMP-2. Finally, the third is the TFLMP-1 with Two-step loss interval filtering technique, which is labelled TFLMP-3.

The assumption in this study is that the LERs observed at receivers are independent of the sender's rate [9], where it is determine by the aggregate data flow behaviour of the bottleneck link. Based on the assumption, we set fixed sending and reception rates at 200 Kbps for all TFLMP implementations. The sending rate is distributed across 3 layers. The rate for each layer is set according to the layering scheme employed by the TFLMPs, and the cumulative rate of all layers is 200 Kbps. This serves our needs very well, since all TFLMPs under study used the same sending and reception rate.

All TFLMP implementations are run under three layering schemes. The first layering scheme is rate multiplier (*m*) =1, the second is *m*=1.3, and finally *m*=2. The use of three layering schemes (rate multipliers) is to observe the effect of the layers' size on the precision and stability of LER and TCP-compatible rate estimations. We estimate loss interval size using the technique similar in [8], and all of loss interval samples are averaged using the technique similar in [3].

A well-known dumbbell topology is used as depicted in Fig. 1. The network bandwidth is shared between 1 TFLMP and 127 TCP connections. This represent a high level statistical multiplexing, in which environment the TCP-equation model should perform well [7]. The bottleneck link between router R_1 and R_2 is configured to have a propagation delay of 20 ms and a bandwidth of 25.6 Mbps (theoretical fair bandwidth share of 200 Kbps per-data-flow). All access links have a delay of 2 ms, and are sufficiently provisioned to ensure that packet drops due to congestion only occur at the bottleneck link.

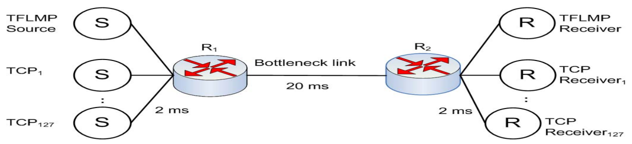

Fig. 1. Simulation topology

We used DVMRP [10] routing protocol at all routers. Droptail and RED queuing policies with buffer size of two bandwidth delay products are used in the experiments. Constant bit rate (CBR) is used as the TFLMP data source, and we set the packet size of all data flows to 1000 bytes. For the TCP data flows we use New TCP Reno, and to avoid the influence of the maximum window, we set max-window to 4000 packets.

We start the multicast source at time zero and its sinks at 3 seconds later. In order to avoid synchronisation, all TCP sessions start at between 3 and 4 seconds using a random number generator (RNG seeds). Each scenario is run 20 times for a total duration of 500 seconds.

VII. RESULTS

Our analysis is based on the trace data produced by the simulations. We ignore the data for the first 100 seconds of each simulation, and measure mean, CoV and variability of the estimated LERs and TCP-compatible rates for the 101^{st} second to 499^{th} second of the simulation. Results are averaged for all 20 simulation runs.

Table I shows the result of LER estimations under Droptail gateway. For all rate multipliers, the average estimated LERs of the TFLMP-1 are the lowest, while the average estimated LERs of the TFLMP-3 are the highest. On the other hand, the CoVs and variabilities of the TFLMP-3 are the lowest, while the CoVs and variabilities of the TFLMP-1 are the highest. These indicate that Two-step loss interval filtering technique results in smoother estimated LERs.

TABLE I
MEAN, COEFFICIENT OF VARIATION AND VARIABILITY OF LER UNDER DROPTAIL GATEWAY

Rate Multiplier	TFLMP Implementations	Average LER	CoV (%)	Variability (%)
m=1	TFLMP-1	0.0597	37.79	9.57
	TFLMP-2	0.0661	31.14	5.82
	TFLMP-3	0.0683	26.68	5.28
m=1.3	TFLMP-1	0.0481	32.41	9.60
	TFLMP-2	0.0522	32.15	6.40
	TFLMP-3	0.0534	30.47	5.74
m=2	TFLMP-1	0.0680	33.16	7.21
	TFLMP-2	0.0711	28.89	4.67
	TFLMP-3	0.0719	26.78	4.14

Table II shows the result of the TCP-compatible rate estimations under Droptail gateway. For all rate multipliers, the average estimated TCP-compatible rates of the TFLMP-1 are the highest, while the average estimated TCP-compatible rates of the TFLMP-3 are the lowest. For *m*=1 and *m*=2, the CoVs of the estimated TCP-compatible rates of the TFLMP-3 are the lowest, while the CoVs of the estimated TCP-compatible rates of the TFLMP-1 are the highest. For *m*=1.3, the CoV of the estimated TCP-compatible rates of the TFLMP-1 is the lowest, while the CoV of the estimated TCP-compatible rates of the TFLMP-2 is the highest. Though the CoV of the estimated TCP-compatible rates of the TFLMP-1 is higher than the CoV of the estimated TCP-compatible rate of the TFLMP-3, the standard deviation of the TFLMP-1 is higher than the standard deviation of the TFLMP-2 – their standard deviations are 43851 bps and 41731 bps respectively. For all rate multipliers, the variabilities of the estimated TCP-compatible rates of the TFLMP-3 are the lowest, while the variabilities of the estimated TCP-compatible rates of the TFLMP-1 are the highest.

TABLE II
MEAN, COEFFICIENT OF VARIATION AND VARIABILITY OF ESTIMATED TCP-COMPATIBLE RATES UNDER DROPTAIL GATEWAY

Rate Multiplier	TFLMP Implementations	Average TCP-compatible rate	CoV (%)	Variability (%)
m=1	TFLMP-1	190771	34.02	5.97
	TFLMP-2	172661	29.12	4.03
	TFLMP-3	165543	24.54	3.75
m=1.3	TFLMP-1	209626	20.90	5.85
	TFLMP-2	199509	21.98	4.23
	TFLMP-3	195879	21.24	3.85
m=2	TFLMP-1	171661	34.18	4.99
	TFLMP-2	163146	29.96	3.45
	TFLMP-3	160000	26.91	3.13

Table III shows the result of the LER estimations under RED gateway. For all TFLMP implementations, the average LERs are quite similar. The CoVs and variabilities of the estimated LER of the TFLMP-1 are the highest, while the CoVs and variabilities of the estimated LER of the TFLMP-3 are the lowest.

TABLE III
MEAN, COEFFICIENT OF VARIATION AND VARIABILITY OF LER UNDER RED GATEWAY

Rate Multiplier	TFLMP Implementations	Average LER	CoV (%)	Variability (%)
m=1	TFLMP-1	0.0572	28.34	10.48
	TFLMP-2	0.0583	27.11	6.09
	TFLMP-3	0.0571	24.23	5.26
m=1.3	TFLMP-1	0.0570	29.44	11.02
	TFLMP-2	0.0579	28.20	6.16
	TFLMP-3	0.0568	25.12	5.37
m=2	TFLMP-1	0.0566	27.95	10.55
	TFLMP-2	0.0582	27.21	6.04
	TFLMP-3	0.0570	24.09	5.22

Table IV shows the result of the TCP-compatible rate estimations under RED gateway. For all TFLMP implementations, the average estimated TCP-compatible rates are quite similar. The average TCP-compatible rates of all TFLMP implementations are slightly higher than the theoretical fair bandwidth share. This is expected as the TCP-equation model performs better than the TCP in a high level of statistical multiplexing environment. For all rate multipliers, the CoVs and variabilities of the estimated TCP-compatible rates of the TFLMP-1 are the highest, while the CoVs and variabilities of the estimated TCP-compatible rates of the TFLMP-3 are the lowest.

TABLE IV
MEAN, COEFFICIENT OF VARIATION AND VARIABILITY OF ESTIMATED TCP-COMPATIBLE RATES UNDER RED GATEWAY

Rate Multiplier	TFLMP Implementations	Average TCP-compatible rate	CoV (%)	Variability (%)
m=1	TFLMP-1	224048	18.79	7.67
	TFLMP-2	220767	18.19	5.34
	TFLMP-3	222483	16.14	4.93
m=1.3	TFLMP-1	224264	19.16	7.94
	TFLMP-2	221374	18.37	5.35
	TFLMP-3	222840	16.29	4.97
m=2	TFLMP-1	224630	18.68	7.69
	TFLMP-2	220349	18.22	5.32
	TFLMP-3	221819	16.07	4.95

VIII. DISCUSSION

Two-step loss interval filtering technique results in smoother estimated LERs and TCP-compatible rates, where significant reduction of CoV and variability are observed in nearly all simulations. This is in favour of continuous media applications, which require stable and smooth data transmission. However, under Droptail gateway, Two-step loss interval filtering technique results in a higher average estimated LERs, and consequently lower average TCP-compatible rates are estimated. We plot loss interval distribution and observe a right-skewed loss interval distribution with a long right tail. It is also observed that sometime the loss interval samples are too large, i.e. up to 6 times of the average loss interval size. With Two-step loss interval filtering technique, these extremely high loss interval samples would be discarded. Therefore lower average loss intervals are achieved. This results in lower LER estimations for TFLMP-3.

Under RED gateway, the TFLMP with One-step loss interval filtering technique and the TFLMP with Two-step loss interval filtering technique (TFLMP-2 and TFLMP-3) achieve similar average estimated TCP-compatible rates as the TFLMP with no filtering technique, but with lower CoV and variability of the estimated TCP-compatible rates. The filtering techniques work well under RED gateway due to the random packet drop implemented in RED gateway.

IX. CONCLUSION

We propose a loss event filtering technique that discards too high and too low loss event samples. The results of simulations that use a fixed subscription rate show that the filtering technique significantly reduces CoV and variability of estimated LERs and TCP-compatible rates of TFLMPs. Next we will test the technique in TFLMPs with dynamic subscription rates.

REFERENCES

[1] M. Handley, S. Floyd, J. Pahdye, and J. Widmer, "TCP Friendly Rate Control (TFRC): Protocol Specification," *IETF, RFC 3448*, 2003.

[2] O. Ghazali and S. Hassan, "Loss Interval Distributions of a TCP-Friendly Layered Multicast Protocol," Unpublished paper, 2005.

[3] S. Floyd, M. Handley, J. Padhye, and J. Widmer, "Equation-Based Congestion Control for Unicast Applications," in the Proceeding of ACM Special Interest Group on Data Communications (SIGCOMM 2000), Stockholm, Sweden, 2000.

[4] L. Vicisano, L. Rizzo, and J. Crowcroft, "TCP-like Congestion Control for Layered Multicast Data Transfer," in the Proceeding of IEEE International Conference on Computer Communications (INFOCOM 1998), San Francisco, 1998.

[5] J. Byers, M. Frumin, G. Horn, M. Luby, M. Mitzenmacher, A. Roetter, and W. Shaver, "FLID-DL: Congestion Control for Layered Multicast," in the Proceeding of ACM Second International Workshop on Networked Group Communication (NGC 2000), Palo Alto, 2000.

[6] O. Ghazali and S. Hassan, "Implementation of a TCP-Friendly Layered Multicast Protocol on NS-2 Simulator," Presented at the NS-2 Workshop of the Second Real-Time Technology and Applications Symposium (RENTAS 2004), Kuala Lumpur, 2004.

[7] J. Widmer, "Equation-Based Congestion Control," *Department of Mathematics and Computer Science*. Germany: University of Mannheim, 2000.

[8] M. Luby, V. K. Goyal, S. Skaria, and G. B. Horn, "Wave and Equation Based Rate Control Using Multicast Round Trip Time," in the Proceeding of ACM Special Interest Group on Data Communications (SIGCOMM 2002), Pittsburgh, 2002.

[9] J. Widmer, R. Denda, and M. Mauve, "A Survey of TCP-Friendly Congestion Control," *IEEE Network*, vol. 15, pp. 28-37, 2001.

[10] D. Waitzman, C. Partridge, and S. Deering, "Distance Vector Multicast Routing Protocol," *IETF, RFC 1075*, 1988.